

BFS

소스코드

최백준 choi@startlink.io



C++14

```
1 #include <iostream>
2 #include <queue>
3 #include <tuple>
4 #include <vector>
5 #include <string>
6 #include <cstring>
7 using namespace std;
8 int dx[] = {0,0,1,-1};
9 int dy[] = {1,-1,0,0};
10 int d[10][10][10][10];
11 int hx,hy;
12 pair<bool,bool> simulate(vector<string> &a, int k, int &x, int &y) {
13     if (a[x][y] == '.') return make_pair(false, false);
14     int n = a.size();
15     int m = a[0].size();
16     bool moved = false;
17     while (true) {
18         int nx = x+dx[k];
19         int ny = y+dy[k];
20         if (nx < 0 || nx >= n || ny < 0 || ny >= m) {
21             return make_pair(moved, false);
22         }
23         if (a[nx][ny] == '#') {
24             return make_pair(moved, false);
25         } else if (a[nx][ny] == 'R' || a[nx][ny] == 'B') {
26             return make_pair(moved, false);
27         } else if (a[nx][ny] == '.') {
28             swap(a[nx][ny], a[x][y]);
29             x = nx;
30             y = ny;
31             moved = true;
32         } else if (a[nx][ny] == '0') {
33             a[x][y] = '.';
34             moved = true;
35             return make_pair(moved, true);
36         }
37     }
38     return make_pair(false, false);
39 }
40 pair<bool,bool> next(vector<string> a, int &rx, int &ry, int &bx, int &by, int dir) {
41     a[rx][ry] = 'R';
42     a[bx][by] = 'B';
43     bool hole1=false, hole2=false;
44     while (true) {
45         auto p1 = simulate(a, dir, rx, ry);
46         auto p2 = simulate(a, dir, bx, by);
47         if (p1.first == false && p2.first == false) {
48             break;
49         }
50         if (p1.second) hole1 = true;
51         if (p2.second) hole2 = true;
52     }
53     return make_pair(hole1, hole2);
54 }
55 int main() {
56     int n, m;
57     cin >> n >> m;
58     vector<string> a(n);
59     for (int i=0; i<n; i++) {
60         cin >> a[i];
61     }
62     int ans = -1;
63     queue<tuple<int,int,int,int>> q;
64     int rx,ry,bx,by;
65     for (int i=0; i<n; i++) {
66         for (int j=0; j<m; j++) {
67             if (a[i][j] == '0') {
68                 hx = i; hy = j;
69             } else if (a[i][j] == 'R') {
70                 rx = i; ry = j;
71                 a[i][j] = '.';
72             } else if (a[i][j] == 'B') {
73                 bx = i; by = j;
74                 a[i][j] = '.';
75             }
76         }
77     }
78     memset(d,-1,sizeof(d));
79     q.emplace(rx,ry,bx,by);
80     d[rx][ry][bx][by] = 0;
81     bool found = false;
82     while (!q.empty()) {
83         tie(rx,ry,bx,by) = q.front();
84         q.pop();
85         for (int k=0; k<4; k++) {
86             bool hole1, hole2;
87             int nrx = rx, nry = ry, nbx = bx, nby = by;
88             tie(hole1, hole2) = next(a, nrx, nry, nbx, nby, k);
89             if (hole2) continue;
90             if (hole1) {
91                 found = true;
92                 ans = d[rx][ry][bx][by] + 1;
93                 break;
94             }
95             if (d[nrx][nry][nbx][nby] != -1) continue;
96             q.emplace(nrx,nry,nbx,nby);
97             d[nrx][nry][nbx][nby] = d[rx][ry][bx][by] + 1;
98         }
99         if (found) {
100             break;
101         }
102     }
103     cout << ans << '\n';
104     return 0;
105 }
106
```

결과	메모리	시간	코드 길이
맞았습니다!!	2032 KB	0 ms	3041 B

Java

```
1 import java.util.*;
2 class Pair {
3     boolean first, second;
4     Pair(boolean first, boolean second) {
5         this.first = first;
6         this.second = second;
7     }
8 }
9 class Point {
10     int x, y;
11     Point(int x, int y) {
12         this.x = x;
13         this.y = y;
14     }
15 }
16 class Points {
17     int rx, ry, bx, by;
18     Points(int rx, int ry, int bx, int by) {
19         this.rx = rx;
20         this.ry = ry;
21         this.bx = bx;
22         this.by = by;
23     }
24 }
25 public class Main {
26     static final int[] dx = {0,0,1,-1};
27     static final int[] dy = {1,-1,0,0};
28     static int hx = 0;
29     static int hy = 0;
30     static Pair simulate(char[][] a, int k, Point p) {
31         int x = p.x;
32         int y = p.y;
33         if (a[x][y] == '.') return new Pair(false, false);
34         int n = a.length;
35         int m = a[0].length;
36         boolean moved = false;
37         while (true) {
38             int nx = x+dx[k];
39             int ny = y+dy[k];
40             if (nx < 0 || nx >= n || ny < 0 || ny >= m) {
41                 p.x = x; p.y = y;
42                 return new Pair(moved, false);
43             }
44             char ch = a[nx][ny];
45             if (ch == '#') {
46                 p.x = x; p.y = y;
47                 return new Pair(moved, false);
48             } else if (ch == 'R' || ch == 'B') {
49                 p.x = x; p.y = y;
50                 return new Pair(moved, false);
51             } else if (ch == '.') {
52                 char temp = a[nx][ny];
53                 a[nx][ny] = a[x][y];
54                 a[x][y] = temp;
55                 x = nx;
56                 y = ny;
57                 moved = true;
58             } else if (ch == '0') {
59                 a[x][y] = '.';
60                 moved = true;
61                 p.x = x; p.y = y;
62                 return new Pair(moved, true);
63             }
64         }
65     }
66     static Pair next(char[][] a, Points p, int dir) {
67         int n = a.length;
68         int m = a[0].length;
69         char[][] b = new char[n][m];
70         for (int i=0; i<n; i++) {
71             for (int j=0; j<m; j++) {
72                 b[i][j] = a[i][j];
73             }
74         }
75         int rx = p.rx, ry = p.ry, bx = p.bx, by = p.by;
76         b[rx][ry] = 'R';
77         b[bx][by] = 'B';
78         Point red = new Point(rx, ry);
79         Point blue = new Point(bx, by);
80         boolean hole1 = false, hole2 = false;
81         while (true) {
82             Pair p1 = simulate(b, dir, red);
83             Pair p2 = simulate(b, dir, blue);
84             if (p1.first == false && p2.first == false) {
85                 break;
86             }
87             if (p1.second) hole1 = true;
88             if (p2.second) hole2 = true;
89         }
90         p.rx = red.x; p.ry = red.y; p.bx = blue.x; p.by = blue.y;
91         return new Pair(hole1, hole2);
92     }
93     public static void main(String args[]) {
94         Scanner sc = new Scanner(System.in);
95         int n = sc.nextInt();
96         int m = sc.nextInt();
97         char[][] a = new char[n][m];
98         for (int i=0; i<n; i++) {
99             String s = sc.next();
100             a[i] = s.toCharArray();
101         }
102         int ans = -1;
103         Queue<Points> q = new LinkedList<>();
104         int rx=0,ry=0,bx=0,by=0;
105         for (int i=0; i<n; i++) {
106             for (int j=0; j<m; j++) {
107                 if (a[i][j] == '0') {
108                     hx = i; hy = j;
109                 } else if (a[i][j] == 'R') {
110                     rx = i; ry = j;
111                     a[i][j] = '.';
112                 } else if (a[i][j] == 'B') {
113                     bx = i; by = j;
114                     a[i][j] = '.';
115                 }
116             }
117         }
118         int[][][] d = new int[n][m][n][m];
119         for (int i=0; i<n; i++) {
120             for (int j=0; j<m; j++) {
121                 for (int k=0; k<n; k++) {
122                     Arrays.fill(d[i][j][k],-1);
123                 }
124             }
125         }
126         q.add(new Points(rx, ry, bx, by));
127         d[rx][ry][bx][by] = 0;
128         boolean found = false;
129         while (!q.isEmpty()) {
130             Points p = q.remove();
131             rx = p.rx;
132             ry = p.ry;
133             bx = p.bx;
134             by = p.by;
135             for (int k=0; k<4; k++) {
136                 boolean hole1 = false, hole2 = false;
137                 int nrx = rx, nry = ry, nbx = bx, nby = by;
138                 temp = new Points(nrx,nry,nbx,nby);
139                 Pair p = next(a,temp,k);
140                 nrx = temp.rx;
141                 nry = temp.ry;
142                 nbx = temp.bx;
143                 nby = temp.by;
144                 hole1 = p.first;
145                 hole2 = p.second;
146                 if (hole2) continue;
147                 if (hole1) {
148                     found = true;
149                     ans = d[rx][ry][bx][by] + 1;
150                     break;
151                 }
152                 if (d[nrx][nry][nbx][nby] != -1) continue;
153                 q.add(new Points(nrx,nry,nbx,nby));
154                 d[nrx][nry][nbx][nby] = d[rx][ry][bx][by] + 1;
155             }
156             if (found) {
157                 break;
158             }
159         }
160         System.out.println(ans);
161     }
162 }
163
```

결과	메모리	시간	코드 길이
맞았습니다!!	12184 KB	120 ms	5138 B

C++14

```
1 #include <iostream>
2 #include <stack>
3 #include <queue>
4 using namespace std;
5 int n;
6 int a[500][500][2];
7 bool check[500][500];
8 pair<int,int> from[500][500];
9 int dist[500][500];
10 int dx0[6] = {-1, -1, 0, 0, 1, 1};
11 int dy0[6] = {-1, 0, -1, 1, -1, 0};
12 int dx1[6] = {-1, -1, 0, 0, 1, 1};
13 int dy1[6] = {0, 1, -1, 1, 0, 1};
14 bool ok(int x, int y) {
15     if (x < 0 || x >= n) return false;
16     if (x % 2 == 0) {
17         return y >= 0 && y < n;
18     } else {
19         return y >= 0 && y < n-1;
20     }
21 }
22 bool go(int x1, int y1, int x2, int y2) {
23     if (x1 == x2) {
24         if (y1 < y2) {
25             return a[x1][y1][1] == a[x2][y2][0];
26         } else {
27             return a[x1][y1][0] == a[x2][y2][1];
28         }
29     } else {
30         if (x1%2 == 0) {
31             if (y1 == y2) {
32                 return a[x1][y1][1] == a[x2][y2][0];
33             } else {
34                 return a[x1][y1][0] == a[x2][y2][1];
35             }
36         } else {
37             if (y1 == y2) {
38                 return a[x1][y1][0] == a[x2][y2][1];
39             } else {
40                 return a[x1][y1][1] == a[x2][y2][0];
41             }
42         }
43     }
44 }
45 int num(int x, int y) {
46     int ans = x/2*(n*2-1);
47     if (x%2 == 1) {
48         ans += n;
49     }
50     ans += y+1;
51     return ans;
52 }
53 int main() {
54     cin >> n;
55     for (int i=0; i<n; i++) {
56         for (int j=0; j<n-1; j++) {
57             cin >> a[i][j][0];
58             cin >> a[i][j][1];
59         }
60         if (i%2 == 0) {
61             cin >> a[i][n-1][0];
62             cin >> a[i][n-1][1];
63         }
64     }
65     queue<pair<int,int>> q;
66     check[0][0] = true;
67     dist[0][0] = 1;
68     q.push(make_pair(0,0));
69     while (!q.empty()) {
70         int x = q.front().first;
71         int y = q.front().second;
72         q.pop();
73         for (int k=0; k<6; k++) {
74             int nx, ny;
75             if (x%2 == 0) {
76                 nx = x+dx0[k];
77                 ny = y+dy0[k];
78             } else {
79                 nx = x+dx1[k];
80                 ny = y+dy1[k];
81             }
82             if (ok(nx, ny) == false) continue;
83             if (go(x, y, nx, ny) == false) continue;
84             if (check[nx][ny] == true) continue;
85             check[nx][ny] = true;
86             dist[nx][ny] = dist[x][y] + 1;
87             from[nx][ny] = make_pair(x, y);
88             q.push(make_pair(nx, ny));
89         }
90     }
91     int x = n-1;
92     int y = n-1;
93     while (check[x][y] == false) {
94         y -= 1;
95         if (y < 0) {
96             x -= 1;
97             y = n-1;
98             if (x % 2 == 1) {
99                 y -= 1;
100             }
101         }
102     }
103     cout << dist[x][y] << '\n';
104     stack<pair<int,int>> s;
105     while (!(x==0 && y==0)) {
106         s.push(make_pair(x,y));
107         auto p = from[x][y];
108         x = p.first;
109         y = p.second;
110     }
111     s.push(make_pair(x, y));
112     while (!s.empty()) {
113         cout << num(s.top().first, s.top().second) << ' ';
114         s.pop();
115     }
116     cout << '\n';
117     return 0;
118 }
```

결과	메모리	시간	코드 길이
맞았습니다!!!	7116 KB	104 ms	2886 B

Java

```
1 import java.util.*;
2 class Pair {
3     int first;
4     int second;
5     Pair(int first, int second) {
6         this.first = first;
7         this.second = second;
8     }
9 }
10 public class Main {
11     static int[] dx = {1,-1,0,0};
12     static int[] dy = {0,0,1,-1};
13     static int n;
14     static int[][] a = new int[500][500][2];
15     static boolean[][] check = new boolean[500][500];
16     static Pair[][] from = new Pair[500][500];
17     static int[][] dist = new int[500][500];
18     static int[] dx0 = {-1, -1, 0, 0, 1, 1};
19     static int[] dy0 = {-1, 0, -1, 1, -1, 0};
20     static int[] dx1 = {-1, -1, 0, 0, 1, 1};
21     static int[] dy1 = {0, 1, -1, 1, 0, 1};
22     static boolean ok(int x, int y) {
23         if (x < 0 || x >= n) return false;
24         if (x % 2 == 0) {
25             return y >= 0 && y < n;
26         } else {
27             return y >= 0 && y < n-1;
28         }
29     }
30     static boolean go(int x1, int y1, int x2, int y2) {
31         if (x1 == x2) {
32             if (y1 < y2) {
33                 return a[x1][y1][1] == a[x2][y2][0];
34             } else {
35                 return a[x1][y1][0] == a[x2][y2][1];
36             }
37         } else {
38             if (x1%2 == 0) {
39                 if (y1 == y2) {
40                     return a[x1][y1][1] == a[x2][y2][0];
41                 } else {
42                     return a[x1][y1][0] == a[x2][y2][1];
43                 }
44             } else {
45                 if (y1 == y2) {
46                     return a[x1][y1][0] == a[x2][y2][1];
47                 } else {
48                     return a[x1][y1][1] == a[x2][y2][0];
49                 }
50             }
51         }
52     }
53     static int num(int x, int y) {
54         int ans = x/2*(n*2-1);
55         if (x%2 == 1) {
56             ans += n;
57         }
58         ans += y+1;
59         return ans;
60     }
61     public static void main(String[] args) {
62         Scanner sc = new Scanner(System.in);
63         n = sc.nextInt();
64         for (int i=0; i<n; i++) {
65             for (int j=0; j<n-1; j++) {
66                 a[i][j][0] = sc.nextInt();
67                 a[i][j][1] = sc.nextInt();
68             }
69             if (i%2 == 0) {
70                 a[i][n-1][0] = sc.nextInt();
71                 a[i][n-1][1] = sc.nextInt();
72             }
73         }
74         Queue<Pair> q = new LinkedList<Pair>();
75         check[0][0] = true;
76         dist[0][0] = 1;
77         q.add(new Pair(0,0));
78         while (!q.isEmpty()) {
79             Pair p = q.remove();
80             int x = p.first;
81             int y = p.second;
82             for (int k=0; k<6; k++) {
83                 int nx, ny;
84                 if (x%2 == 0) {
85                     nx = x+dx0[k];
86                     ny = y+dy0[k];
87                 } else {
88                     nx = x+dx1[k];
89                     ny = y+dy1[k];
90                 }
91                 if (ok(nx, ny) == false) continue;
92                 if (go(x, y, nx, ny) == false) continue;
93                 if (check[nx][ny] == true) continue;
94                 check[nx][ny] = true;
95                 dist[nx][ny] = dist[x][y] + 1;
96                 from[nx][ny] = new Pair(x, y);
97                 q.add(new Pair(nx, ny));
98             }
99         }
100         int x = n-1;
101         int y = n-1;
102         while (check[x][y] == false) {
103             y -= 1;
104             if (y < 0) {
105                 x -= 1;
106                 y = n-1;
107                 if (x % 2 == 1) {
108                     y -= 1;
109                 }
110             }
111         }
112         System.out.println(dist[x][y]);
113         Stack<Pair> s = new Stack<Pair>();
114         while (!(x==0 && y==0)) {
115             s.push(new Pair(x,y));
116             Pair p = from[x][y];
117             x = p.first;
118             y = p.second;
119         }
120         s.push(new Pair(x, y));
121         while (!s.isEmpty()) {
122             System.out.print(num(s.peek().first, s.peek().second) + " ");
123             s.pop();
124         }
125         System.out.println();
126     }
127 }
```

결과	메모리	시간	코드 길이
맞았습니다!!!	168196 KB	1392 ms	3803 B

C++14

```
1 #include <iostream>
2 #include <string>
3 #include <queue>
4 using namespace std;
5 int n, m;
6 string a[250];
7 bool check[250][250];
8 int d[250*250][2];
9 int dx[] = {0,0,1,-1};
10 int dy[] = {1,-1,0,0};
11 void bfs(int sx, int sy, int cnt) {
12     queue<pair<int,int>> q;
13     q.push(make_pair(sx,sy));
14     check[sx][sy] = true;
15     while (!q.empty()) {
16         int x = q.front().first;
17         int y = q.front().second;
18         q.pop();
19         if (a[x][y] == 'v') {
20             d[cnt][0] += 1;
21         } else if (a[x][y] == 'o') {
22             d[cnt][1] += 1;
23         }
24         for (int k=0; k<4; k++) {
25             int nx = x+dx[k];
26             int ny = y+dy[k];
27             if (nx < 0 || nx >= n || ny < 0 || ny >= m) continue;
28             if (a[nx][ny] == '#') continue;
29             if (check[nx][ny]) continue;
30             q.push(make_pair(nx,ny));
31             check[nx][ny] = true;
32         }
33     }
34 }
35 int main() {
36     cin >> n >> m;
37     for (int i=0; i<n; i++) {
38         cin >> a[i];
39     }
40     int cnt = 0;
41     for (int i=0; i<n; i++) {
42         for (int j=0; j<m; j++) {
43             if (a[i][j] != '#' && check[i][j] == false) {
44                 cnt += 1;
45                 bfs(i, j, cnt);
46             }
47         }
48     }
49     int v = 0;
50     int o = 0;
51     for (int i=1; i<=cnt; i++) {
52         if (d[i][0] >= d[i][1]) {
53             v += d[i][0];
54         } else {
55             o += d[i][1];
56         }
57     }
58     cout << o << ' ' << v << '\n';
59     return 0;
60 }
```

결과	메모리	시간	코드 길이
맞았습니다!!	2680 KB	4 ms	1433 B

Java

```
1 import java.util.*;
2 class Pair {
3     int first;
4     int second;
5     Pair(int first, int second) {
6         this.first = first;
7         this.second = second;
8     }
9 }
10 public class Main {
11     static int[] dx = {1,-1,0,0};
12     static int[] dy = {0,0,1,-1};
13     static int n, m;
14     static String[] a = new String[250];
15     static boolean[][] check = new boolean[250][250];
16     static int[][] d = new int[250*250][2];
17     static void bfs(int sx, int sy, int cnt) {
18         Queue<Pair> q = new LinkedList<Pair>();
19         q.add(new Pair(sx,sy));
20         check[sx][sy] = true;
21         while (!q.isEmpty()) {
22             Pair p = q.remove();
23             int x = p.first;
24             int y = p.second;
25             if (a[x].charAt(y) == 'v') {
26                 d[cnt][0] += 1;
27             } else if (a[x].charAt(y) == 'o') {
28                 d[cnt][1] += 1;
29             }
30             for (int k=0; k<4; k++) {
31                 int nx = x+dx[k];
32                 int ny = y+dy[k];
33                 if (nx < 0 || nx >= n || ny < 0 || ny >= m) continue;
34                 if (a[nx].charAt(ny) == '#') continue;
35                 if (check[nx][ny]) continue;
36                 q.add(new Pair(nx,ny));
37                 check[nx][ny] = true;
38             }
39         }
40     }
41     public static void main(String[] args) {
42         Scanner sc = new Scanner(System.in);
43         n = sc.nextInt();
44         m = sc.nextInt();
45         for (int i=0; i<n; i++) {
46             a[i] = sc.next();
47         }
48         int cnt = 0;
49         for (int i=0; i<n; i++) {
50             for (int j=0; j<m; j++) {
51                 if (a[i].charAt(j) != '#' && check[i][j] == false) {
52                     cnt += 1;
53                     bfs(i, j, cnt);
54                 }
55             }
56         }
57         int v = 0;
58         int o = 0;
59         for (int i=1; i<=cnt; i++) {
60             if (d[i][0] >= d[i][1]) {
61                 v += d[i][0];
62             } else {
63                 o += d[i][1];
64             }
65         }
66         System.out.println(o + " " + v);
67     }
68 }
```

결과	메모리	시간	코드 길이
맞았습니다!!	22008 KB	240 ms	1996 B

C++14

```
1 #include <iostream>
2 #include <queue>
3 using namespace std;
4 int dist[1000001];
5 bool check[1000001];
6 int main() {
7     int f, s, g, u, d;
8     cin >> f >> s >> g >> u >> d;
9     queue<int> q;
10    q.push(s);
11    check[s] = true;
12    while (!q.empty()) {
13        int now = q.front();
14        q.pop();
15        if (now + u <= f && check[now+u] == false) {
16            dist[now+u] = dist[now] + 1;
17            check[now+u] = true;
18            q.push(now+u);
19        }
20        if (now - d >= 1 && check[now-d] == false) {
21            dist[now-d] = dist[now] + 1;
22            check[now-d] = true;
23            q.push(now-d);
24        }
25    }
26    if (check[g]) {
27        cout << dist[g] << '\n';
28    } else {
29        cout << "use the stairs\n";
30    }
31    return 0;
32 }
```

결과	메모리	시간	코드 길이
맞았습니다!!	6872 KB	8 ms	777 B

Java

```
1 import java.util.*;
2 public class Main {
3     public static void main(String args[]) {
4         Scanner sc = new Scanner(System.in);
5         int f = sc.nextInt();
6         int s = sc.nextInt();
7         int g = sc.nextInt();
8         int u = sc.nextInt();
9         int d = sc.nextInt();
10        int[] dist = new int[f+1];
11        boolean[] check = new boolean[f+1];
12        Queue<Integer> q = new LinkedList<>();
13        q.add(s);
14        check[s] = true;
15        while (!q.isEmpty()) {
16            int now = q.remove();
17            if (now + u <= f && check[now+u] == false) {
18                dist[now+u] = dist[now] + 1;
19                check[now+u] = true;
20                q.add(now+u);
21            }
22            if (now - d >= 1 && check[now-d] == false) {
23                dist[now-d] = dist[now] + 1;
24                check[now-d] = true;
25                q.add(now-d);
26            }
27        }
28        if (check[g]) {
29            System.out.println(dist[g]);
30        } else {
31            System.out.println("use the stairs");
32        }
33    }
34 }
35
```

결과	메모리	시간	코드 길이
맞았습니다!!	50116 KB	196 ms	1074 B

C++14

```
1 #include <iostream>
2 #include <queue>
3 using namespace std;
4 bool check[1501][1501];
5 int sum;
6 void go(int x, int y) {
7     if (check[x][y]) return;
8     check[x][y] = true;
9     int a[3] = {x, y, sum-x-y};
10    for (int i=0; i<3; i++) {
11        for (int j=0; j<3; j++) {
12            if (a[i] < a[j]) {
13                int b[3] = {x, y, sum-x-y};
14                b[i] += a[i];
15                b[j] -= a[i];
16                go(b[0], b[1]);
17            }
18        }
19    }
20 }
21 int main() {
22     int x, y, z;
23     cin >> x >> y >> z;
24     sum = x + y + z;
25     if (sum % 3 != 0) {
26         cout << 0 << '\n';
27         return 0;
28     }
29     go(x, y);
30     if (check[sum/3][sum/3]) {
31         cout << 1 << '\n';
32     } else {
33         cout << 0 << '\n';
34     }
35     return 0;
36 }
37
```

(x,y)

(BFS)

DFS

결과	메모리	시간	코드 길이
맞았습니다!!	32688 KB	52 ms	783 B

Java

```
1 import java.util.*;
2 public class Main {
3     public static int sum = 0;
4     public static boolean[][] check = new boolean[1501][1501];
5     public static void go(int x, int y) {
6         if (check[x][y]) return;
7         check[x][y] = true;
8         int[] a = {x, y, sum-x-y};
9         for (int i=0; i<3; i++) {
10             for (int j=0; j<3; j++) {
11                 if (a[i] < a[j]) {
12                     int[] b = {x, y, sum-x-y};
13                     b[i] += a[i];
14                     b[j] -= a[i];
15                     go(b[0], b[1]);
16                 }
17             }
18         }
19     }
20 }
21 public static void main(String args[]) {
22     Scanner sc = new Scanner(System.in);
23     int x = sc.nextInt();
24     int y = sc.nextInt();
25     int z = sc.nextInt();
26     sum = x+y+z;
27     if (sum % 3 != 0) {
28         System.out.println(0);
29         System.exit(0);
30     }
31     go(x, y);
32     if (check[sum/3][sum/3]) {
33         System.out.println(1);
34     } else {
35         System.out.println(0);
36     }
37 }
38 }
39
```

결과	메모리	시간	코드 길이
맞았습니다!!	90036 KB	396 ms	1077 B

C++14

```
1 #include <iostream>
2 #include <queue>
3 #include <cstdio>
4 #include <tuple>
5 #include <cstring>
6 using namespace std;
7 int a[1000][1000];
8 int d[1000][1000][11];
9 int dx[] = {0, 0, 1, -1};
10 int dy[] = {1, -1, 0, 0};
11 int main() {
12     int n, m, l;
13     scanf("%d %d %d",&n,&m,&l);
14     for (int i=0; i<n; i++) {
15         for (int j=0; j<m; j++) {
16             scanf("%1d",&a[i][j]);
17         }
18     }
19     queue<tuple<int,int,int>> q;
20     d[0][0][0] = 1;
21     q.push(make_tuple(0,0,0));
22     while (!q.empty()) {
23         int x, y, z;
24         tie(x,y,z) = q.front(); q.pop();
25         for (int k=0; k<4; k++) {
26             int nx = x+dx[k];
27             int ny = y+dy[k];
28             if (nx < 0 || nx >= n || ny < 0 || ny >= m) continue;
29             if (a[nx][ny] == 0 && d[nx][ny][z] == 0) {
30                 d[nx][ny][z] = d[x][y][z] + 1;
31                 q.push(make_tuple(nx,ny,z));
32             }
33             if (z+1 <= l && a[nx][ny] == 1 && d[nx][ny][z+1] == 0) {
34                 d[nx][ny][z+1] = d[x][y][z] + 1;
35                 q.push(make_tuple(nx,ny,z+1));
36             }
37         }
38     }
39     int ans = -1;
40     for (int i=0; i<=l; i++) {
41         if (d[n-1][m-1][i] == 0) continue;
42         if (ans == -1) {
43             ans = d[n-1][m-1][i];
44         } else if (ans > d[n-1][m-1][i]) {
45             ans = d[n-1][m-1][i];
46         }
47     }
48     cout << ans << '\n';
49     return 0;
50 }
51
```

결과	메모리	시간	코드 길이
맞았습니다!!!	48996 KB	332 ms	1406 B

Java

```

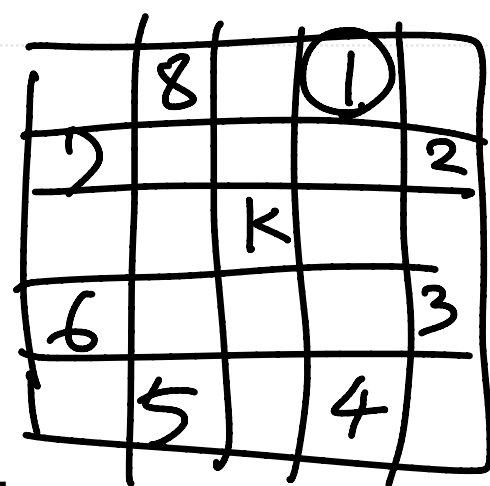
1 import java.util.*;
2 class Pair {
3     int x, y, z;
4     Pair(int x, int y, int z) {
5         this.x = x;
6         this.y = y;
7         this.z = z;
8     }
9 }
10 public class Main {
11     public static int[] dx = {1, -1, 0, 0};
12     public static int[] dy = {0, 0, 1, -1};
13     public static void main(String[] args) {
14         Scanner sc = new Scanner(System.in);
15         int n = sc.nextInt();
16         int m = sc.nextInt();
17         int l = sc.nextInt();
18         sc.nextLine();
19         int[][] a = new int[n][m];
20         int[][][] d = new int[n][m][l+1];
21         for (int i=0; i<n; i++) {
22             String s = sc.nextLine();
23             for (int j=0; j<m; j++) {
24                 a[i][j] = s.charAt(j) - '0';
25             }
26         }
27         d[0][0][0] = 1;
28         Queue<Pair> q = new LinkedList<Pair>();
29         q.offer(new Pair(0, 0, 0));
30         while (!q.isEmpty()) {
31             Pair p = q.remove();
32             int x = p.x;
33             int y = p.y;
34             int z = p.z;
35             for (int k=0; k<4; k++) {
36                 int nx = x+dx[k];
37                 int ny = y+dy[k];
38                 if (nx < 0 || nx >= n || ny < 0 || ny >= m) continue;
39                 if (a[nx][ny] == 0 && d[nx][ny][z] == 0) {
40                     d[nx][ny][z] = d[x][y][z] + 1;
41                     q.offer(new Pair(nx, ny, z));
42                 }
43                 if (z+1 <= l && a[nx][ny] == 1 && d[nx][ny][z+1] == 0) {
44                     d[nx][ny][z+1] = d[x][y][z] + 1;
45                     q.offer(new Pair(nx, ny, z+1));
46                 }
47             }
48         }
49         int ans = -1;
50         for (int i=0; i<=l; i++) {
51             if (d[n-1][m-1][i] == 0) continue;
52             if (ans == -1) {
53                 ans = d[n-1][m-1][i];
54             } else if (ans > d[n-1][m-1][i]) {
55                 ans = d[n-1][m-1][i];
56             }
57         }
58         System.out.println(ans);
59     }
60 }
61
62

```

결과	메모리	시간	코드 길이
맞았습니다!!!	202164 KB	1676 ms	1921 B

C++14

```
1 #include <iostream>
2 #include <tuple>
3 #include <queue>
4 #include <cstring>
5 using namespace std;
6 int dx[] = {0,0,1,-1,-2,-1,1,2,2,1,-1,-2};
7 int dy[] = {1,-1,0,0,1,2,2,1,-1,-2,-2,-1};
8 int cost[] = {0,0,0,0,1,1,1,1,1,1,1,1};
9 int a[200][200];
10 int d[200][200][31];
11 int main() {
12     int l;
13     cin >> l;
14     int n, m;
15     cin >> m >> n;
16     for (int i=0; i<n; i++) {
17         for (int j=0; j<m; j++) {
18             cin >> a[i][j];
19         }
20     }
21     memset(d,-1,sizeof(d));
22     queue<tuple<int,int,int>> q;
23     d[0][0][0] = 0;
24     q.push(make_tuple(0,0,0));
25     while (!q.empty()) {
26         int x, y, c;
27         tie(x,y,c) = q.front();
28         q.pop();
29         for (int k=0; k<12; k++) {
30             int nx = x+dx[k];
31             int ny = y+dy[k];
32             int nc = c+cost[k];
33             if (0 <= nx && nx < n && 0 <= ny && ny < m) {
34                 if (a[nx][ny] == 1) continue;
35                 if (nc <= l) {
36                     if (d[nx][ny][nc] == -1) {
37                         d[nx][ny][nc] = d[x][y][c] + 1;
38                         q.push(make_tuple(nx,ny,nc));
39                     }
40                 }
41             }
42         }
43     }
44     int ans = -1;
45     for (int i=0; i<=l; i++) {
46         if (d[n-1][m-1][i] == -1) continue;
47         if (ans == -1 || ans > d[n-1][m-1][i]) {
48             ans = d[n-1][m-1][i];
49         }
50     }
51     cout << ans << '\n';
52     return 0;
53 }
```



4
4

(0,0)

(4+8)
0,4 4

C++14

```
1 #include <iostream>
2 #include <queue>
3 #include <vector>
4 #include <string>
5 using namespace std;
6 int dx[] = {0,0,1,-1};
7 int dy[] = {1,-1,0,0};
8 bool can(bool blind, char from, char to) {
9     if (from == to) return true;
10    if (blind) {
11        if (from == 'R' && to == 'G') return true;
12        if (from == 'G' && to == 'R') return true;
13    }
14    return false;
15 }
16 int go(vector<string> &a, bool blind = false) {
17     int n = a.size();
18     vector<vector<bool>> check(n, vector<bool>(n,false));
19     int ans = 0;
20     for (int i=0; i<n; i++) {
21         for (int j=0; j<n; j++) {
22             if (check[i][j] == false) {
23                 ans += 1;
24                 queue<pair<int,int>> q;
25                 q.push(make_pair(i,j));
26                 check[i][j] = true;
27                 while (!q.empty()) {
28                     int x = q.front().first;
29                     int y = q.front().second;
30                     q.pop();
31                     for (int k=0; k<4; k++) {
32                         int nx = x+dx[k];
33                         int ny = y+dy[k];
34                         if (0 <= nx && nx < n && 0 <= ny && ny < n) {
35                             if (check[nx][ny]) continue;
36                             if (can(blind,a[x][y],a[nx][ny])) {
37                                 check[nx][ny] = true;
38                                 q.push(make_pair(nx,ny));
39                             }
40                         }
41                     }
42                 }
43             }
44         }
45     }
46     return ans;
47 }
48 int main() {
49     int n;
50     cin >> n;
51     vector<string> a(n);
52     for (int i=0; i<n; i++) {
53         cin >> a[i];
54     }
55     cout << go(a) << ' ' << go(a,true) << '\n';
56     return 0;
57 }
58
```

결과	메모리	시간	코드 길이
맞았습니다!!	1992 KB	0 ms	1727 B

Java

```
1 import java.util.*;
2 public class Main {
3     static final int[] dx = {0,0,1,-1};
4     static final int[] dy = {1,-1,0,0};
5     static boolean can(boolean blind, char from, char to) {
6         if (from == to) return true;
7         if (blind) {
8             if (from == 'R' && to == 'G') return true;
9             if (from == 'G' && to == 'R') return true;
10        }
11        return false;
12    }
13    static int go(String[] a, boolean blind) {
14        int n = a.length;
15        boolean[][] check = new boolean[n][n];
16        int ans = 0;
17        for (int i=0; i<n; i++) {
18            for (int j=0; j<n; j++) {
19                if (check[i][j] == false) {
20                    ans += 1;
21                    Queue<Integer> q = new LinkedList<>();
22                    q.add(i); q.add(j);
23                    check[i][j] = true;
24                    while (!q.isEmpty()) {
25                        int x = q.remove();
26                        int y = q.remove();
27                        for (int k=0; k<4; k++) {
28                            int nx = x+dx[k];
29                            int ny = y+dy[k];
30                            if (0 <= nx && nx < n && 0 <= ny && ny < n) {
31                                if (check[nx][ny]) continue;
32                                if (can(blind,a[x].charAt(y),a[nx].charAt(ny))) {
33                                    check[nx][ny] = true;
34                                    q.add(nx); q.add(ny);
35                                }
36                            }
37                        }
38                    }
39                }
40            }
41        }
42        return ans;
43    }
44    public static void main(String args[]) {
45        Scanner sc = new Scanner(System.in);
46        int n = sc.nextInt();
47        String[] a = new String[n];
48        for (int i=0; i<n; i++) {
49            a[i] = sc.next();
50        }
51        System.out.println(go(a,false) + " " + go(a,true));
52    }
53 }
54
```

결과	메모리	시간	코드 길이
맞았습니다!!	13028 KB	164 ms	1949 B

C++14

```
1 #include <iostream>
2 #include <queue>
3 using namespace std;
4 int n, m;
5 int a[50][50];
6 int d[50][50];
7 int room[50*50];
8 int dx[] = {0,-1,0,1};
9 int dy[] = {-1,0,1,0};
10 int bfs(int x, int y, int rooms) {
11     queue<pair<int,int>> q;
12     q.push(make_pair(x,y));
13     d[x][y] = rooms;
14     int cnt = 0;
15     while (!q.empty()) {
16         x = q.front().first;
17         y = q.front().second;
18         q.pop();
19         cnt += 1;
20         for (int k=0; k<4; k++) {
21             int nx = x+dx[k];
22             int ny = y+dy[k];
23             if (nx < 0 || nx >= n || ny < 0 || ny >= m) continue;
24             if (d[nx][ny] != 0) continue;
25             if (a[x][y] & (1<<k)) continue;
26             q.push(make_pair(nx,ny));
27             d[nx][ny] = rooms;
28         }
29     }
30     return cnt;
31 }
32 int main() {
33     cin >> m >> n;
34     for (int i=0; i<n; i++) {
35         for (int j=0; j<m; j++) {
36             cin >> a[i][j];
37         }
38     }
39     int rooms = 0;
40     for (int i=0; i<n; i++) {
41         for (int j=0; j<m; j++) {
42             if (d[i][j] == 0) {
43                 rooms += 1;
44                 room[rooms] = bfs(i, j, rooms);
45             }
46         }
47     }
48     cout << rooms << '\n';
49     int ans = 0;
50     for (int i=1; i<=rooms; i++) {
51         if (ans < room[i]) {
52             ans = room[i];
53         }
54     }
55     cout << ans << '\n';
56     ans = 0;
57     for (int i=0; i<n; i++) {
58         for (int j=0; j<m; j++) {
59             int x = i;
60             int y = j;
61             for (int k=0; k<4; k++) {
62                 int nx = x+dx[k];
63                 int ny = y+dy[k];
64                 if (nx < 0 || nx >= n || ny < 0 || ny >= m) continue;
65                 if (d[nx][ny] == d[x][y]) continue;
66                 if (a[x][y] & (1<<k)) {
67                     if (ans < room[d[x][y]]+room[d[nx][ny]]) {
68                         ans = room[d[x][y]]+room[d[nx][ny]];
69                     }
70                 }
71             }
72         }
73     }
74     cout << ans << '\n';
75     return 0;
76 }
77
```

Handwritten notes and diagrams:

- Diagram of a 3x3 grid with a center cell (1,1) and its four neighbors (0,1), (2,1), (1,0), (1,2). Arrows indicate movement directions.
- Text: "몇번 방문?" (How many times visited?) with an arrow pointing to the center cell.
- Text: "7번 방문?" (7 times visited?) with an arrow pointing to the center cell.
- Text: "(x,y)" with an arrow pointing to the center cell.
- Text: "cnt += 1;" with an arrow pointing to the line.
- Text: "1 2 4 8" with an arrow pointing to the line.
- Text: "2 2 2 2" with an arrow pointing to the line.
- Text: "2" with an arrow pointing to the line.
- Text: "(x,y)" and "nx,ny" with arrows pointing to the variables.
- Text: "d[nx][ny] == d[x][y]" with an arrow pointing to the line.
- Text: "a[x][y] & (1<<k)" with an arrow pointing to the line.

결과	메모리	시간	코드 길이
맞았습니다!!	2016 KB	0 ms	1997 B

Java

```
1 import java.util.*;
2 class Pair {
3     int x;
4     int y;
5     Pair(int x, int y) {
6         this.x = x;
7         this.y = y;
8     }
9 }
10 public class Main {
11     static int n;
12     static int m;
13     static int[][] a = new int[50][50];
14     static int[][] d = new int[50][50];
15     static int[] room = new int[50*50+1];
16     static int[] dx = {0,-1,0,1};
17     static int[] dy = {-1,0,1,0};
18     static int bfs(int sx, int sy, int rooms) {
19         Queue<Pair> q = new LinkedList<Pair>();
20         q.add(new Pair(sx, sy));
21         d[sx][sy] = rooms;
22         int cnt = 0;
23         while (!q.isEmpty()) {
24             Pair p = q.remove();
25             int x = p.x;
26             int y = p.y;
27             cnt += 1;
28             for (int k=0; k<4; k++) {
29                 int nx = x+dx[k];
30                 int ny = y+dy[k];
31                 if (nx < 0 || nx >= n || ny < 0 || ny >= m) continue;
32                 if (d[nx][ny] != 0) continue;
33                 if ((a[x][y] & (1<<k)) > 0) continue;
34                 q.add(new Pair(nx, ny));
35                 d[nx][ny] = rooms;
36             }
37         }
38         return cnt;
39     }
40     public static void main(String[] args) {
41         Scanner sc = new Scanner(System.in);
42         m = sc.nextInt();
43         n = sc.nextInt();
44         for (int i=0; i<n; i++) {
45             for (int j=0; j<m; j++) {
46                 a[i][j] = sc.nextInt();
47             }
48         }
49         int rooms = 0;
50         for (int i=0; i<n; i++) {
51             for (int j=0; j<m; j++) {
52                 if (d[i][j] == 0) {
53                     rooms += 1;
54                     room[rooms] = bfs(i, j, rooms);
55                 }
56             }
57         }
58         System.out.println(rooms);
59         int ans = 0;
60         for (int i=1; i<=rooms; i++) {
61             if (ans < room[i]) {
62                 ans = room[i];
63             }
64         }
65         System.out.println(ans);
66         ans = 0;
67         for (int i=0; i<n; i++) {
68             for (int j=0; j<m; j++) {
69                 int x = i;
70                 int y = j;
71                 for (int k=0; k<4; k++) {
72                     int nx = x+dx[k];
73                     int ny = y+dy[k];
74                     if (nx < 0 || nx >= n || ny < 0 || ny >= m) continue;
75                     if (d[nx][ny] == d[x][y]) continue;
76                     if ((a[x][y] & (1<<k)) > 0) {
77                         if (ans < room[d[x][y]]+room[d[nx][ny]]) {
78                             ans = room[d[x][y]]+room[d[nx][ny]];
79                         }
80                     }
81                 }
82             }
83         }
84         System.out.println(ans);
85     }
86 }
87
```

결과	메모리	시간	코드 길이
맞았습니다!!	16840 KB	188 ms	2630 B

C++14

```
1 #include <iostream>
2 #include <vector>
3 #include <queue>
4 #include <string>
5 using namespace std;
6 string a[1500];
7 int dx[] = {0,0,1,-1};
8 int dy[] = {1,-1,0,0};
9 bool wcheck[1500][1500];
10 bool scheck[1500][1500];
11 int main() {
12     int n, m;
13     cin >> n >> m;
14     int sx,sy,ex,ey;
15     sx=sy=ex=ey=-1;
16     queue<pair<int,int>> swan, nswan, water, nwater;
17     for (int i=0; i<n; i++) {
18         cin >> a[i];
19         for (int j=0; j<m; j++) {
20             if (a[i][j] == 'L') {
21                 if (sx == -1) {
22                     sx = i;
23                     sy = j;
24                 } else {
25                     ex = i;
26                     ey = j;
27                 }
28                 a[i][j] = '.';
29             }
30             if (a[i][j] == '.') {
31                 water.push(make_pair(i,j));
32                 wcheck[i][j] = true;
33             }
34         }
35     }
36     swan.push(make_pair(sx,sy));
37     scheck[sx][sy] = true;
38     for (int i=0; i<n; i++) {
39         while (!water.empty()) {
40             int x = water.front().first;
41             int y = water.front().second;
42             water.pop();
43             a[x][y] = '.';
44             for (int k=0; k<4; k++) {
45                 int nx = x+dx[k];
46                 int ny = y+dy[k];
47                 if (nx < 0 || nx >= n || ny < 0 || ny >= m) continue;
48                 if (wcheck[nx][ny]) continue;
49                 if (a[nx][ny] == '.') {
50                     water.push(make_pair(nx,ny));
51                     wcheck[nx][ny] = true;
52                 } else {
53                     nwater.push(make_pair(nx,ny));
54                     wcheck[nx][ny] = true;
55                 }
56             }
57         }
58         while (!swan.empty()) {
59             int x = swan.front().first;
60             int y = swan.front().second;
61             swan.pop();
62             for (int k=0; k<4; k++) {
63                 int nx = x+dx[k];
64                 int ny = y+dy[k];
65                 if (nx < 0 || nx >= n || ny < 0 || ny >= m) continue;
66                 if (scheck[nx][ny]) continue;
67                 if (a[nx][ny] == '.') {
68                     swan.push(make_pair(nx,ny));
69                     scheck[nx][ny] = true;
70                 } else {
71                     nswan.push(make_pair(nx,ny));
72                     scheck[nx][ny] = true;
73                 }
74             }
75         }
76         if (scheck[ex][ey]) {
77             cout << i << '\n';
78             break;
79         }
80         water = nwater;
81         swan = nswan;
82         nwater = queue<pair<int,int>>();
83         nswan = queue<pair<int,int>>();
84     }
85     return 0;
86 }
```

결과

메모리

시간

코드 길이

맞았습니다!!

17852 KB

232 ms

2501 B

Java

```
1 import java.util.*;
2 class Pair {
3     int first;
4     int second;
5     Pair(int first, int second) {
6         this.first = first;
7         this.second = second;
8     }
9 }
10 public class Main {
11     static int[] dx = {1,-1,0,0};
12     static int[] dy = {0,0,1,-1};
13     public static void main(String[] args) {
14         Scanner sc = new Scanner(System.in);
15         int n = sc.nextInt();
16         int m = sc.nextInt();
17         int sx,sy,ex,ey;
18         sx=sy=ex=ey=-1;
19         Queue<Pair> swan, nswan, water, nwater;
20         swan = new LinkedList<Pair>();
21         nswan = new LinkedList<Pair>();
22         water = new LinkedList<Pair>();
23         nwater = new LinkedList<Pair>();
24         char[][] a = new char[n][m];
25         boolean[][] wcheck = new boolean[n][m];
26         boolean[][] scheck = new boolean[n][m];
27         for (int i=0; i<n; i++) {
28             a[i] = sc.next().toArray();
29             for (int j=0; j<m; j++) {
30                 if (a[i][j] == 'L') {
31                     if (sx == -1) {
32                         sx = i;
33                         sy = j;
34                     } else {
35                         ex = i;
36                         ey = j;
37                     }
38                     a[i][j] = '.';
39                 }
40                 if (a[i][j] == '.') {
41                     water.add(new Pair(i, j));
42                     wcheck[i][j] = true;
43                 }
44             }
45         }
46         swan.add(new Pair(sx,sy));
47         scheck[sx][sy] = true;
48         for (int i=0;; i++) {
49             while (!water.isEmpty()) {
50                 Pair p = water.remove();
51                 int x = p.first;
52                 int y = p.second;
53                 a[x][y] = '.';
54                 for (int k=0; k<4; k++) {
55                     int nx = x+dx[k];
56                     int ny = y+dy[k];
57                     if (nx < 0 || nx >= n || ny < 0 || ny >= m) continue;
58                     if (wcheck[nx][ny]) continue;
59                     if (a[nx][ny] == '.') {
60                         water.add(new Pair(nx,ny));
61                         wcheck[nx][ny] = true;
62                     } else {
63                         nwater.add(new Pair(nx,ny));
64                         wcheck[nx][ny] = true;
65                     }
66                 }
67             }
68             while (!swan.isEmpty()) {
69                 Pair p = swan.remove();
70                 int x = p.first;
71                 int y = p.second;
72                 for (int k=0; k<4; k++) {
73                     int nx = x+dx[k];
74                     int ny = y+dy[k];
75                     if (nx < 0 || nx >= n || ny < 0 || ny >= m) continue;
76                     if (scheck[nx][ny]) continue;
77                     if (a[nx][ny] == '.') {
78                         swan.add(new Pair(nx,ny));
79                         scheck[nx][ny] = true;
80                     } else {
81                         nswan.add(new Pair(nx,ny));
82                         scheck[nx][ny] = true;
83                     }
84                 }
85             }
86             if (scheck[ex][ey]) {
87                 System.out.println(i);
88                 break;
89             }
90             water = nwater;
91             swan = nswan;
92             nwater = new LinkedList<Pair>();
93             nswan = new LinkedList<Pair>();
94         }
95     }
96 }
```

결과

메모리

시간

코드 길이

맞았습니다!!

146036 KB

1224 ms

3206 B

C++14

```
1 #include <iostream>
2 #include <map>
3 #include <queue>
4 #include <array>
5 using namespace std;
6 int main() {
7     array<string,3> s;
8     for (int i=0; i<3; i++) {
9         int cnt;
10        cin >> cnt;
11        if (cnt > 0) {
12            cin >> s[i];
13        } else {
14            s[i] = "";
15        }
16    }
17    int cnt[3] = {0, 0, 0};
18    for (int i=0; i<3; i++) {
19        for (int j=0; j<s[i].length(); j++) {
20            cnt[s[i][j]-'A'] += 1;
21        }
22    }
23    map<array<string,3>, int> d;
24    queue<array<string,3>> q;
25    q.push(s);
26    d[s] = 0;
27    while (!q.empty()) {
28        auto now = q.front();
29        q.pop();
30        for (int i=0; i<3; i++) {
31            for (int j=0; j<3; j++) {
32                if (i == j) continue;
33                if (now[i].length() == 0) continue;
34                array<string,3> next(now);
35                next[j].push_back(next[i].back());
36                next[i].pop_back();
37                if (d.count(next) == 0) {
38                    d[next] = d[now] + 1;
39                    q.push(next);
40                }
41            }
42        }
43    }
44    array<string,3> ans;
45    for (int i=0; i<3; i++) {
46        for (int j=0; j<cnt[i]; j++) {
47            ans[i] += (char)('A' + i);
48        }
49    }
50    cout << d[ans] << '\n';
51    return 0;
52 }
```

A→B B→A
A→C I→J

결과

메모리

시간

코드 길이

맞았습니다!!

45592 KB

1424 ms

1317 B

C++14

```
1 #include <iostream>
2 #include <tuple>
3 #include <queue>
4 #include <string>
5 #include <set>
6 using namespace std;
7 const long long limit = 1000000000LL;
8 int main() {
9     long long s, t;
10    cin >> s >> t;
11    set<long long> check;
12    queue<pair<long long, string>> q;
13    q.push(make_pair(s, ""));
14    check.insert(s);
15    while (!q.empty()) {
16        long long x;
17        string str;
18        tie(x, str) = q.front(); q.pop();
19        if (x == t) {
20            if (str.length() == 0) {
21                str = "0";
22            }
23            cout << str << '\n';
24            return 0;
25        }
26        if (0 <= x*x && x*x <= limit && check.count(x*x) == 0) {
27            q.push(make_pair(x*x, str+"*"));
28            check.insert(x*x);
29        }
30        if (0 <= x+x && x+x <= limit && check.count(x+x) == 0) {
31            q.push(make_pair(x+x, str+"+"));
32            check.insert(x+x);
33        }
34        if (0 <= x-x && x-x <= limit && check.count(x-x) == 0) {
35            q.push(make_pair(x-x, str+"-"));
36            check.insert(x-x);
37        }
38        if (x != 0 && 0 <= x/x && x/x <= limit && check.count(x/x) == 0) {
39            q.push(make_pair(x/x, str+"/"));
40            check.insert(x/x);
41        }
42    }
43    cout << -1 << '\n';
44    return 0;
45 }
```

결과	메모리	시간	코드 길이
맞았습니다!!	1992 KB	0 ms	1289 B

Java

```
1 import java.util.*;
2 public class Main {
3     final static long limit = 1000000000L;
4     public static void main(String args[]) {
5         Scanner sc = new Scanner(System.in);
6         long s = sc.nextLong();
7         long t = sc.nextLong();
8         Queue<Long> q = new LinkedList<Long>();
9         Queue<String> qs = new LinkedList<String>();
10        HashSet<Long> check = new HashSet<Long>();
11        q.add(s);
12        qs.add("");
13        check.add(s);
14        while (!q.isEmpty()) {
15            long x = q.remove();
16            String str = qs.remove();
17            if (x == t) {
18                if (str.length() == 0) {
19                    str = "0";
20                }
21                System.out.println(str);
22                System.exit(0);
23            }
24            if (0 <= x*x && x*x <= limit && check.contains(x*x) == false) {
25                q.add(x*x);
26                qs.add(str+"*");
27                check.add(x*x);
28            }
29            if (0 <= x+x && x+x <= limit && check.contains(x+x) == false) {
30                q.add(x+x);
31                qs.add(str+"+");
32                check.add(x+x);
33            }
34            if (0 <= x-x && x-x <= limit && check.contains(x-x) == false) {
35                q.add(x-x);
36                qs.add(str+"-");
37                check.add(x-x);
38            }
39            if (x != 0 && 0 <= x/x && x/x <= limit && check.contains(x/x) == false) {
40                q.add(x/x);
41                qs.add(str+"/");
42                check.add(x/x);
43            }
44        }
45        System.out.println(-1);
46    }
47 }
```

결과	메모리	시간	코드 길이
맞았습니다!!	12848 KB	128 ms	1593 B

C++14

```
1 #include <iostream>
2 #include <vector>
3 #include <queue>
4 using namespace std;
5 int dx[] = {0,0,1,-1};
6 int dy[] = {1,-1,0,0};
7 int main() {
8     int n;
9     cin >> n;
10    vector<string> s(n);
11    vector<vector<int>>> b(n, vector<int>(n));
12    vector<pair<int,int>> v;
13    int start=-1, end=-1;
14    for (int i=0; i<n; i++) {
15        cin >> s[i];
16        for (int j=0; j<n; j++) {
17            if (s[i][j] == '#') {
18                if (start == -1) {
19                    start = v.size();
20                } else {
21                    end = v.size();
22                }
23                v.push_back(make_pair(i,j));
24                b[i][j] = v.size()-1;
25            } else if (s[i][j] == '!') {
26                v.push_back(make_pair(i,j));
27                b[i][j] = v.size()-1;
28            }
29        }
30    }
31    int m = v.size();
32    vector<vector<bool>>> a(m, vector<bool>(m, false));
33    for (int i=0; i<v.size(); i++) {
34        for (int k=0; k<4; k++) {
35            int x = v[i].first+dx[k];
36            int y = v[i].second+dy[k];
37            while (0 <= x && x < n && 0 <= y && y < n) {
38                if (s[x][y] == '*') break;
39                if (s[x][y] == '!' || s[x][y] == '#') {
40                    a[i][b[x][y]] = true;
41                }
42                x += dx[k];
43                y += dy[k];
44            }
45        }
46    }
47    queue<int> q;
48    vector<int> dist(m, -1);
49    q.push(start);
50    dist[start] = 0;
51    while (!q.empty()) {
52        int now = q.front();
53        q.pop();
54        for (int i=0; i<m; i++) {
55            if (a[now][i] && dist[i] == -1) {
56                dist[i] = dist[now]+1;
57                q.push(i);
58            }
59        }
60    }
61    cout << dist[end]-1 << "\n";
62    return 0;
63 }
64
```

결과	메모리	시간	코드 길이
맞았습니다!!	2128 KB	0 ms	1767 B

Java

```
1 import java.util.*;
2 class Pair {
3     int x, y;
4     Pair(int x, int y) {
5         this.x = x;
6         this.y = y;
7     }
8 }
9 public class Main {
10     static final int[] dx = {0,0,1,-1};
11     static final int[] dy = {1,-1,0,0};
12     public static void main(String[] args) {
13         Scanner sc = new Scanner(System.in);
14         int n = sc.nextInt();
15         String[] s = new String[n];
16         int[][] b = new int[n][n];
17         ArrayList<Pair> v = new ArrayList<>();
18         int start=-1, end=-1;
19         for (int i=0; i<n; i++) {
20             s[i] = sc.next();
21             for (int j=0; j<n; j++) {
22                 char ch = s[i].charAt(j);
23                 if (ch == '#') {
24                     if (start == -1) {
25                         start = v.size();
26                     } else {
27                         end = v.size();
28                     }
29                     v.add(new Pair(i,j));
30                     b[i][j] = v.size()-1;
31                 } else if (ch == '!') {
32                     v.add(new Pair(i,j));
33                     b[i][j] = v.size()-1;
34                 }
35             }
36         }
37         int m = v.size();
38         boolean[][] a = new boolean[m][m];
39         for (int i=0; i<v.size(); i++) {
40             for (int k=0; k<4; k++) {
41                 int x = v.get(i).x + dx[k];
42                 int y = v.get(i).y + dy[k];
43                 while (0 <= x && x < n && 0 <= y && y < n) {
44                     char ch = s[x].charAt(y);
45                     if (ch == '*') break;
46                     if (ch == '!' || ch == '#') {
47                         a[i][b[x][y]] = true;
48                     }
49                     x += dx[k];
50                     y += dy[k];
51                 }
52             }
53         }
54         Queue<Integer> q = new LinkedList<>();
55         int[] dist = new int[m];
56         Arrays.fill(dist,-1);
57         q.add(start);
58         dist[start] = 0;
59         while (!q.isEmpty()) {
60             int now = q.remove();
61             for (int i=0; i<m; i++) {
62                 if (a[now][i] && dist[i] == -1) {
63                     dist[i] = dist[now]+1;
64                     q.add(i);
65                 }
66             }
67         }
68         System.out.println(dist[end]-1);
69     }
70 }
71
```

결과	메모리	시간	코드 길이
맞았습니다!!	13532 KB	152 ms	2240 B

C++14

```

1 #include <iostream>
2 #include <queue>
3 #include <cstring>
4 #include <tuple>
5 using namespace std;
6 int n, m;
7 int a[10][10];
8 int dist[10][10][20];
9 int dx[] = {0,0,1,-1};
10 int dy[] = {1,-1,0,0};
11 int bfs() {
12     memset(dist,-1,sizeof(dist));
13     queue<tuple<int,int,int>> q;
14     q.push(make_tuple(0,0,0));
15     dist[0][0][0] = 0;
16     while (!q.empty()) {
17         int x,y,t;
18         tie(x,y,t) = q.front();
19         q.pop();
20         if (a[x][y] >= 2 && t % a[x][y] != 0) {
21             int nt = (t+1)%a[x][y];
22             if (dist[x][y][nt] == -1) {
23                 dist[x][y][nt] = dist[x][y][t] + 1;
24                 q.push(make_tuple(x,y,nt));
25             }
26         } else {
27             for (int k=0; k<4; k++) {
28                 int nx = x+dx[k];
29                 int ny = y+dy[k];
30                 if (0 <= nx && nx < n && 0 <= ny && ny < n) {
31                     if (a[x][y] <= 2 && a[nx][ny] <= 2) continue;
32                     if (a[nx][ny] >= 1) {
33                         int nt = (dist[x][y][t]+1)%a[nx][ny];
34                         if (dist[nx][ny][nt] == -1) {
35                             dist[nx][ny][nt] = dist[x][y][t] + 1;
36                             q.push(make_tuple(nx,ny,nt));
37                         }
38                     }
39                 }
40             }
41         }
42     }
43     int ans = -1;
44     for (int i=0; i<20; i++) {
45         if (dist[n-1][n-1][i] == -1) continue;
46         if (ans == -1 || ans > dist[n-1][n-1][i]) {
47             ans = dist[n-1][n-1][i];
48         }
49     }
50     return ans;
51 }
52 bool can(int i, int j) {
53     bool garo = false;
54     if (j-1 >= 0 && a[i][j-1] == 0) garo = true;
55     if (j+1 < n && a[i][j+1] == 0) garo = true;
56     bool sero = false;
57     if (i-1 >= 0 && a[i-1][j] == 0) sero = true;
58     if (i+1 < n && a[i+1][j] == 0) sero = true;
59     return !(garo && sero);
60 }
61 int main() {
62     cin >> n >> m;
63     for (int i=0; i<n; i++) {
64         for (int j=0; j<n; j++) {
65             cin >> a[i][j];
66         }
67     }
68     int ans = -1;
69     for (int i=0; i<n; i++) {
70         for (int j=0; j<n; j++) {
71             if (a[i][j] == 0 && can(i, j)) {
72                 a[i][j] = m;
73                 int now = bfs();
74                 if (now != -1) {
75                     if (ans == -1 || ans > now) {
76                         ans = now;
77                     }
78                 }
79                 a[i][j] = 0;
80             }
81         }
82     }
83     cout << ans << '\n';
84     return 0;
85 }
86

```

결과	메모리	시간	코드 길이
맞았습니다!!	1996 KB	0 ms	2462 B

Java

```

1 import java.util.*;
2 public class Main {
3     static int[][] a;
4     static int n, m;
5     static int[][][] dist;
6     static final int[] dx = {0,0,1,-1};
7     static final int[] dy = {1,-1,0,0};
8     static int bfs() {
9         for (int i=0; i<n; i++) {
10             for (int j=0; j<n; j++) {
11                 Arrays.fill(dist[i][j],-1);
12             }
13         }
14         Queue<Integer> q = new LinkedList<>();
15         q.add(0); q.add(0); q.add(0);
16         dist[0][0][0] = 0;
17         while (!q.isEmpty()) {
18             int x = q.remove();
19             int y = q.remove();
20             int t = q.remove();
21             if (a[x][y] >= 2 && t % a[x][y] != 0) {
22                 int nt = (t+1)%a[x][y];
23                 if (dist[x][y][nt] == -1) {
24                     dist[x][y][nt] = dist[x][y][t] + 1;
25                     q.add(x); q.add(y); q.add(nt);
26                 }
27             } else {
28                 for (int k=0; k<4; k++) {
29                     int nx = x+dx[k];
30                     int ny = y+dy[k];
31                     if (0 <= nx && nx < n && 0 <= ny && ny < n) {
32                         if (a[x][y] >= 2 && a[nx][ny] >= 2) continue;
33                         if (a[nx][ny] >= 1) {
34                             int nt = (dist[x][y][t]+1)%a[nx][ny];
35                             if (dist[nx][ny][nt] == -1) {
36                                 dist[nx][ny][nt] = dist[x][y][t] + 1;
37                                 q.add(nx); q.add(ny); q.add(nt);
38                             }
39                         }
40                     }
41                 }
42             }
43         }
44         int ans = -1;
45         for (int i=0; i<20; i++) {
46             if (dist[n-1][n-1][i] == -1) continue;
47             if (ans == -1 || ans > dist[n-1][n-1][i]) {
48                 ans = dist[n-1][n-1][i];
49             }
50         }
51         return ans;
52     }
53     static boolean can (int i, int j) {
54         boolean garo = false;
55         if (j-1 >= 0 && a[i][j-1] == 0) garo = true;
56         if (j+1 < n && a[i][j+1] == 0) garo = true;
57         boolean sero = false;
58         if (i-1 >= 0 && a[i-1][j] == 0) sero = true;
59         if (i+1 < n && a[i+1][j] == 0) sero = true;
60         return !(garo && sero);
61     }
62     public static void main(String[] args) {
63         Scanner sc = new Scanner(System.in);
64         n = sc.nextInt();
65         m = sc.nextInt();
66         a = new int[n][n];
67         dist = new int[n][n][20];
68         for (int i=0; i<n; i++) {
69             for (int j=0; j<n; j++) {
70                 a[i][j] = sc.nextInt();
71             }
72         }
73         int ans = -1;
74         for (int i=0; i<n; i++) {
75             for (int j=0; j<n; j++) {
76                 if (a[i][j] == 0 && can(i, j)) {
77                     a[i][j] = m;
78                     int now = bfs();
79                     if (now != -1) {
80                         if (ans == -1 || ans > now) {
81                             ans = now;
82                         }
83                     }
84                     a[i][j] = 0;
85                 }
86             }
87         }
88         System.out.println(ans);
89     }
90 }
91

```

결과	메모리	시간	코드 길이
맞았습니다!!	11772 KB	128 ms	3092 B

끝

코드 플러스

<https://code.plus>

- 슬라이드에 포함된 소스 코드를 보려면 "정보 수정 > 백준 온라인 저지 연동"을 통해 연동한 다음, "백준 온라인 저지"에 로그인해야 합니다.
- 강의 내용에 대한 질문은 코드 플러스의 "질문 게시판"에서 할 수 있습니다.
- 문제와 소스 코드는 슬라이드에 첨부된 링크를 통해서 볼 수 있으며, "백준 온라인 저지"에서 서비스됩니다.
- 슬라이드와 동영상 강의는 코드 플러스 사이트를 통해서만 볼 수 있으며, 동영상 강의의 녹화와 다운로드, 배포와 유통은 저작권법에 의해서 금지되어 있습니다.
- 다른 경로로 이 슬라이드나 동영상 강의를 본 경우에는 codeplus@startlink.io 로 이메일 보내주세요.
- 강의 내용, 동영상 강의, 슬라이드, 첨부되어 있는 소스 코드의 저작권은 스타트링크와 최백준에게 있습니다.